

Prof. Dr.-Ing. Rudolf Nocker

Synchroner Leitungsdecodierer für 1B2B-Leitungscode

Synthese mit Booleschen Funktionstabellen und Automatendiagrammen

„Coded Mark Inversion“ (CMI) und „Manchester“ (MAN) sind häufig verwendete binäre Leitungscode. Die Synthese von Decodierer-Schaltungen kann mit Booleschen Funktionstabellen und Automatendiagrammen erfolgen. Diese Daten lassen sich in ein Logiksynthese-Softwarepaket eingeben und liefern als Ergebnis eine standardisierte Datei zur Programmierung eines geeigneten Logikbausteins. Die einzelnen Entwicklungsschritte von der umgangssprachlichen Problembeschreibung bis zur Programmierung sind hier erläutert.

Das übliche Schema der digitalen Nachrichtenübertragung setzt sich aus der Kettenschaltung von Nachrichtenquelle, Quellencodierer, Kanalcodierer, Leitungscodierer, Übertragungskanal mit Störquelleneinwirkung, Leitungsdecodierer, Kanaldecodierer, Quellendecodierer und Nachrichtensenke zusammen (Bild 1). Der Quellencodierer befreit das Signal von Redundanz und Irrelevanz und erzeugt ein Binärsignal mit möglichst geringer Schrittgeschwindigkeit. Der Kanalcodierer fügt diesem Binärsignal gezielt Redundanz in Form von Sicherungsinformation hinzu. Der Leitungscodierer paßt das binäre Ausgangssignal des Kanalcodierers durch Umsetzung in ein binäres oder (häufig) mehrwertiges Ausgangssignal an die Eigen-

schaften des Übertragungskanals (Bandbreite, Störsignale) an. Die empfangsseitigen Blöcke machen die sendeseitigen Operationen weitgehend rückgängig, soweit dies wegen der Beeinflussung des Signals durch Störsignale und Verzerrungen im Übertragungskanal möglich ist. Der Übertragungsabschnitt aus Leitungscodierer, Übertragungskanal und Leitungsdecodierer heißt auch „Binärkanal“, die Kettenschaltung aller vorangestellten Blöcke „Binärquelle“, die Kettenschaltung aller nachfolgenden Blöcke „Binärsenke“.

Bei einfachen Übertragungssystemen können einzelne Blöcke dieses Schemas entfallen (beispielsweise Kanalcodierer und Kanaldecodierer) oder stark entarten (beispielsweise im Quellencodierer ausschließlich Irrelevanzreduktion durch Bandbegrenzung). Die sendeseitige Leitungscodierung zur Anpassung an die Eigenschaften des gestörten Übertragungskanals (Aussteuerbereich bzw. zulässige Sendeleistung, Bandbreite, Störsignale, Impedanzanpassung) und die zugehörige empfangsseitige Leitungsdecodierung sind auch bei einfachen Übertragungssystemen meist unverzichtbar.

Verschiedene Leitungscode

Der Leitungscodierer formt das binäre Eingangssignal entweder in ein binäres oder in ein mehrwertiges Leitungssignal ohne (oder mit konstantem) Gleichanteil und hohem Taktinformationsgehalt um. Die Zuordnungsvorschrift zwischen binärem Eingangssignal und Leitungssignal heißt „Übertragungscode“ oder „Leitungscode“. Die Parameter des Leitungscodiererausgangssignals (Schrittgeschwindigkeit, Stufenanzahl, Signalhub) sind so gewählt, daß sie möglichst gut zu den zugehörigen Parametern des Übertragungsmediums

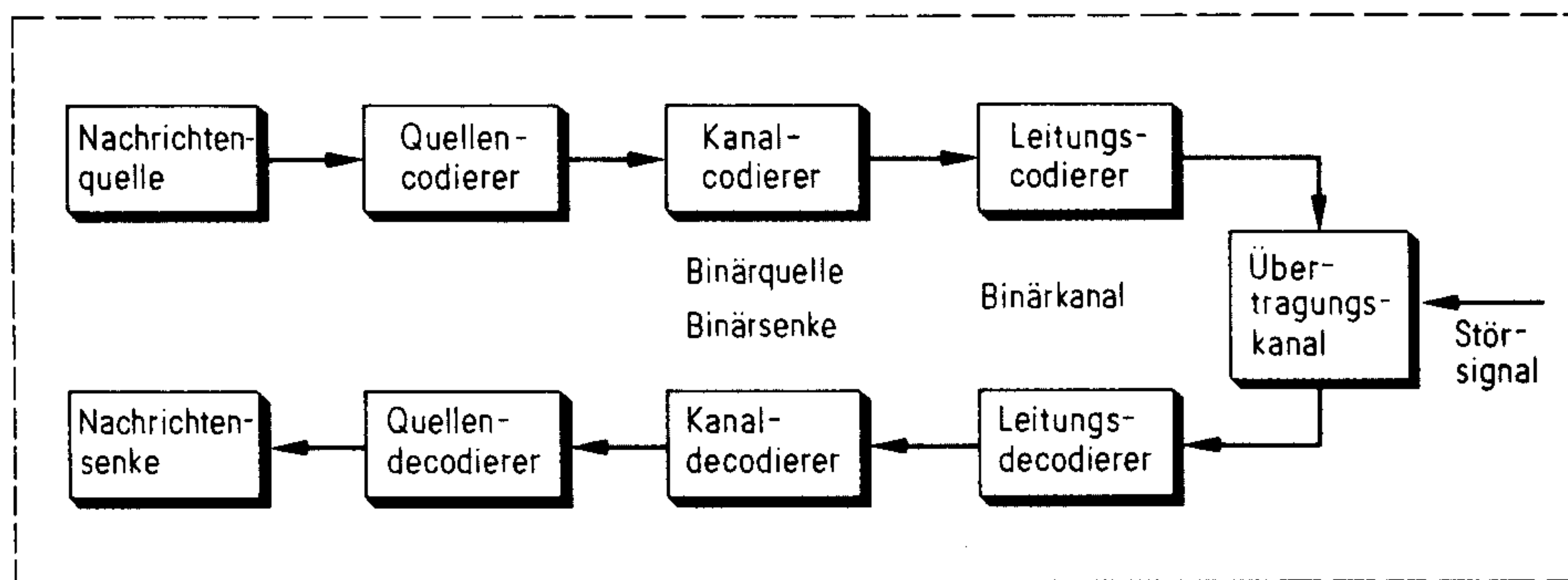


Bild 1. Grundschemata der digitalen Nachrichtenübertragung: Oben Sendezweig, unten Empfangszweig, rechts der Übertragungskanal mit Stör-signal-Beeinflussung.

(Bandbreite, Störsignale, Aussteuerbereich) passen. Der (empfangsseitige) Leitungsdecoder soll aus dem Ausgangssignal des Übertragungskanals eine möglichst unverfälschte Kopie des Binärsignals am Leitungscodierer-Eingang erzeugen. Grundsätzliches über die Prinzipien der Leitungscodierung ist in den Literaturstellen [2, 3] zu finden.

In [4, 5] sind die binären Leitungscodes „Coded Mark Inversion“ (CMI) und „Manchester“ (MAN) sowie die (pseudo)ternären Leitungscodes „Alternate Mark Inversion“ (AMI) und „High Density Bipolarcode 3rd Order“ (HDB3) erläutert; für diese wurde der Entwurf von Codierschaltungen mit programmierbaren Logikbausteinen behandelt. Nachfolgend ist ein synchroner Leitungsdecoder für die binären Leitungscodes CMI und MAN mit derartigen Bausteinen beschrieben.

1B2B-Leitungscodes: Bandbreitenbedarf verdoppelt

Die häufig verwendeten Leitungscodes CMI und MAN sind Codes der 1B2B-Gruppe: Jeder Original-Binärschritt (1B) der Schrittdauer T wird durch zwei Binärschritte (2B) halber Schrittdauer $T/2$ ersetzt. Weitere Codes dieser Gruppe sind beispielsweise in [1] beschrieben. Der MAN-Code heißt auch „Biphase-Level (Bi Φ -L)“, die invertierte Form MAN-II-Code.

Vorteile dieser Codes sind der hohe Taktinformationsgehalt (ermöglicht einfache Taktrückgewinnung) und die maximale Stufenhöhe (ergibt hohe Störsicherheit) bei gegebenem Signalhub. Wesentlicher Nachteil ist der verdoppelte Bandbreitenbedarf gegenüber dem Original-Binärsignal, bedingt durch die halbierte Schrittdauer. Deshalb finden diese Codes bevorzugt dort Einsatz, wo erhöhter Bandbreitenbedarf kostengünstig realisierbar ist; beispielsweise als Schnittstellencodes, als Leitungscodes für Kommunikationsnetze im lokalen Bereich (LANs) und bei optischer Übertragungstechnik.

Tabelle 1 zeigt die Codiervorschriften für den MAN- und den CMI-Code. Beispielsweise wird beim CMI-Code ein Original-Binärschritt des zu codierenden Binärsignals bei 0 in die beiden (Bild-)Binärschritte 01 (sog. Dibit), bei 1 alternierend in die Dibits 00 und 11 codiert. Bei elektrischer Übertragungstechnik wird dann die ausgangsseitige „0“ als negativer Spannungsschritt $-U$, die ausgangsseitige „1“ als positiver Spannungsschritt $+U$ gesendet. Bei optischer Übertragungstechnik wird

Tabelle 1. Codiervorschriften für den MAN- und den CMI-Leitungscode

| BIN | MAN | CMI |
|-----|-----|---------------------|
| 0 | 01 | 01 |
| 1 | 10 | 00, 11 alternierend |

BIN = Eingangs-Binärsignal

MAN = Dibit des „Manchester“-Leitungscodes

CMI = Dibit des „Coded Mark Inversion“-Leitungscodes

mit der ausgangsseitigen Binärfolge ein optisches Sendeelement (Lumineszenzdiode LED, Laserdiode LD) getastet (bei 0 abgeschaltet, bei 1 eingeschaltet).

Die Zuordnung von Teilaufgaben der Signalübertragung zu den Blöcken des eingangs skizzierten Schemas der Signalübertragung (Bild 1) ist mit gewissen Freiheitsgraden möglich. Beispielsweise lassen sich die Sendeschaltung und die Empfangsschaltung (einschließlich Signalregeneration und Taktrückgewinnung) dem Block „Übertragungskanal“ zuordnen, so auch hier. Der Übertragungskanal (bestehend aus Sendeschaltung, Übertragungsmedium und Empfangsschaltung einschließlich Signalregeneration) heißt dann „Digitalkanal“, er antwortet auf ein (im allgemeinen mehrwertiges) digitales Eingangssignal mit einem entsprechenden digitalen Ausgangssignal. Das Digitalkanal-Ausgangssignal weist wegen der Störsignal-Beeinflussung im Übertragungsmedium und in der Empfangsschaltung eine bestimmte Schrittfehlerwahrscheinlichkeit auf. Auf die Realisierung der Empfangsschaltung (einschließlich Signalregeneration mit Taktrückgewinnung) geht dieser Beitrag nicht ein, näheres ist in [1] nachzulesen.

Schaltungskonzept eines 1B2B-Leitungsdecoderers

Nachfolgend ist ein mögliches Schaltungskonzept eines Leitungsdecoderers für 1B2B-Leitungscodes beschrieben und dessen Realisierung am Beispiel eines CMI-Leitungsdecoderes erklärt. Die Schaltung arbeitet vollständig synchron mit einer Maximal-Taktfrequenz, welche gleich der Schrittgeschwindigkeit des Leitungssignals ist. Da keine asynchronen Zeitglieder (Monoflops mit externen Bauelementen) vorhanden sind, kann die Schaltung unverändert für (weitgehend) beliebige Übertragungsgeschwindigkeiten eingesetzt wer-

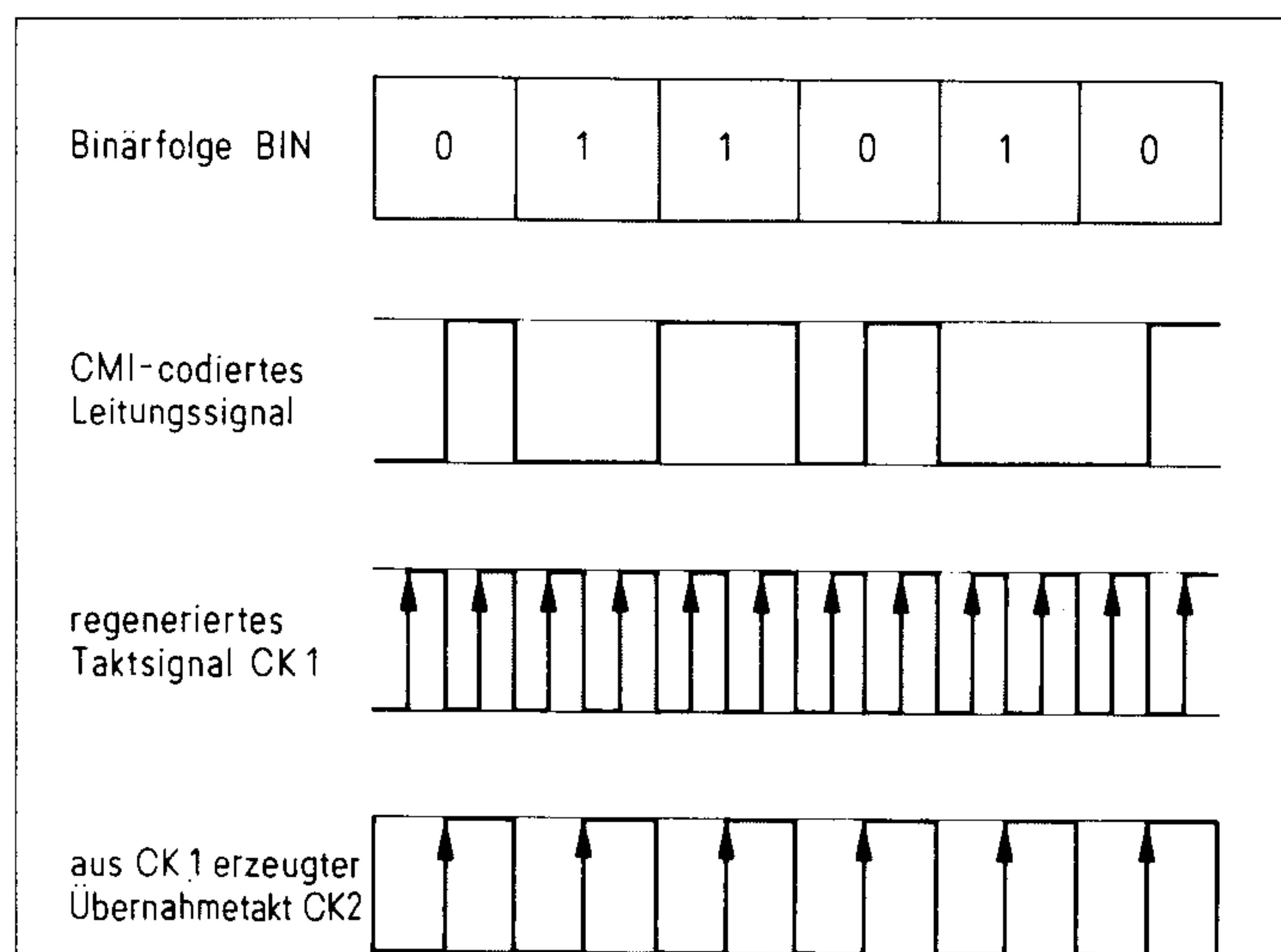


Bild 2. Binärfolge und zugehöriges CMI-Leitungssignal mit den Taktsignalen CK1 und CK2.

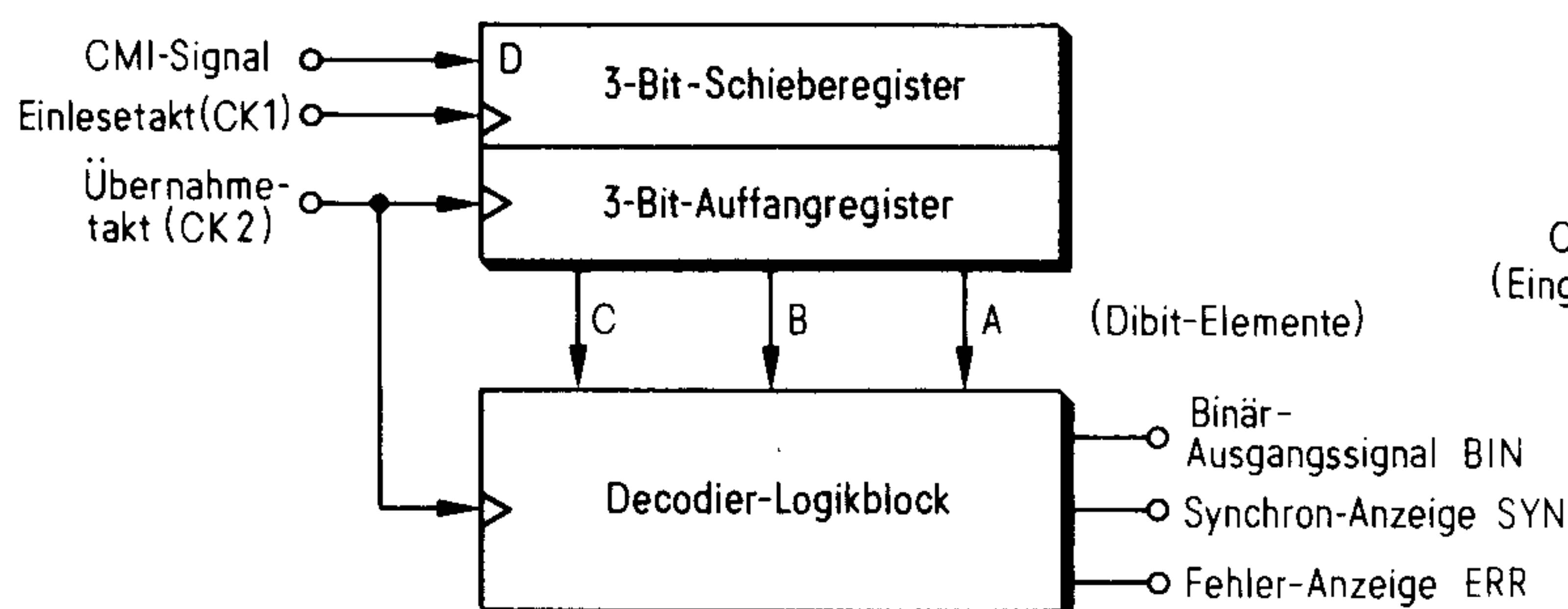


Bild 3. Blockschaltung für einen CMI-Leitungsdecoder.

den. Da externe Bauelemente entfallen, läßt sie sich platzsparend aufbauen und (ggf. zusammen mit synchronen Schaltungsteilen für andere Teilaufgaben) in einem programmierbaren Logikbaustein realisieren. Diese prinzipiellen Vorteile vollsynchroner Digital-schaltungen wiegen stärker als der Nachteil der häufig höheren Schaltungskomplexität gegenüber asynchronen Digital-schaltungen.

Bild 2 zeigt ein zu einer Binärfolge gehöriges CMI-Signal mit dem zugehörigen Taktsignal CK1 am Eingang des Leitungsdecodierers, also nach der Signalregeneration einschließlich der Taktrückgewinnung. Dabei ist fehlerfreie Übertragung angenommen. Empfangsseitig ist zunächst nicht bekannt, welche zwei aufeinanderfolgenden Binärschritte jeweils ein Dibit bilden. Der Leitungsdecodierer muß deshalb zwei Teilaufgaben lösen: 1. Dibit-Synchronisation, 2. Dibit-Decodierung.

Bild 3 zeigt die Leitungsdecodierer-Blockschaltung, bestehend aus einem Serien-Parallel-Umsetzer mit Auffangregister und einem Logikblock. Das serielle CMI-Signal gelangt mit dem Taktsignal CK1 in ein positiv flankengesteuertes 3-Bit-Schieberegister. Nach jeweils zwei Einlesevorgängen fließt der momentane Schiebe-

register-Inhalt mit dem Taktsignal CK2 in ein positiv flankengesteuertes 3-Bit-Auffangregister (Latch). Zu beachten ist hierbei, daß die Übernahme erst eine gewisse Mindestzeit nach der aktiven positiven Flanke von CK1 erfolgen darf. Dies ist gesichert, wenn das Taktsignal CK2 entsprechend *Bild 4* aus dem invertierten Taktsignal CK1 durch einen Zweifach-Takteiler abgeleitet wird. (In *Bild 4* ist das Moore-Automatendiagramm für einen Zweifach-Takteiler eingezeichnet). Diese Teilerschaltung kann je nach Startzustand auch ein Taktsignal CK2 liefern, welches gegenüber dem gezeichneten Verlauf von CK2 genau invertiert ist. Beim beschriebenen Schaltungskonzept ist dies ohne Bedeutung, da das „Aufsynchronisieren“ auf die Dibit-Folge erst im nachfolgenden Schaltungsblock erfolgt. Es sei angemerkt, daß bei speziellen Serien-Parallel-Umsetzer-ICs eine parallele Übernahme auch zulässig ist, wenn die beiden Taktsignale aus einer synchronen Teilerschaltung stammen und damit flankenkoinzident sind (ohne jeden zeitlichen Versatz der aktiven Signalflanken). Die hier beschriebene „sichere“ Methode ist immer zu bevorzugen.

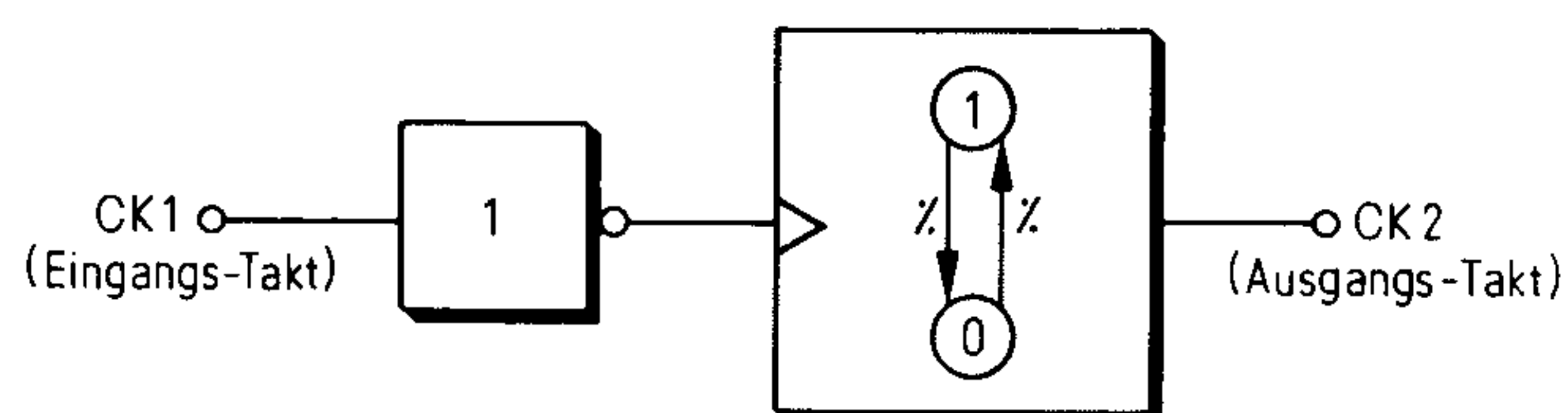


Bild 4. Zweifach-Takteiler; Funktion dargestellt als positiv-flankengesteuerter Moore-Automat.

Das 3-Bit-Register enthält mit Sicherheit ein vollständiges Dibit, entweder in den Speicherstellen AB oder BC. Durch die Serien-Parallel-Umsetzung werden für die nachfolgende Logikschaltung die drei aufeinanderfolgenden Dibit-Elemente A, B, C parallel bereitgestellt. Die Dibit-Synchronisationschaltung stellt fest, ob entweder AB oder BC ein zu decodierendes Dibit enthält. Dies läßt sich daran erkennen, ob die Aufeinanderfolge der Binärschritte A, B, C nach der Codierregel des verwendeten Leitungscodes zulässig ist. Dazu muß beispielsweise AB ein zulässiges Dibit und C ein zulässiges „Folgebit“ (erstes Bit des folgenden Dibit) sein oder BC ein zulässiges Dibit und A ein zulässiges „Vorbit“ (zweites Bit des vorhergehenden Dibit) sein.

Realisierung des Logikblocks

Bild 5 zeigt die Aufteilung des „Logikblocks“ aus *Bild 3* in drei Teilblöcke: a) Synchronisations-Vorverknüpfung, b) Synchronisations-Schaltwerk, c) Decodier-Schaltnetz. Nachfolgend ist beschrieben, wie sich diese Teilblöcke realisieren lassen.

Tabelle 2. Boolesche Funktionstabelle der Synchronisations-Vorverknüpfung für den CMI-Leitungscode

| A | B | C | ABG | BCG | NAB | NBC |
|---|---|---|-----|-----|-----|-----|
| 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 |

A, B, C = Bezeichnung aufeinanderfolgender Dibit-Elemente
ABG = ABGültig (siehe Text) NAB = ABG & /BCG
BCG = BCGültig (siehe Text) NBC = BCG & /ABG

Synchronisations-Vorverknüpfung

Diese prüft, ob entweder in AB oder in BC ein gültiges Dibit des zu decodierenden Leitungssignals enthalten ist. *Tabelle 2* zeigt die Boolesche Funktionstabelle des Blocks „Synchronisations-Vorverknüpfung“ für den CMI-Code. Dabei sind ABG und BCG Zwischengrößen, aus denen durch Nachverknüpfung die Ausgangsgrößen NAB und NBC entstehen. Die Abkürzung ABG bedeutet „AB Gültig“, d.h. AB ist zulässiges Dibit und C zulässiges Folgebit. Die Abkürzung BCG bedeutet „BC Gültig“, d.h. BC ist zulässiges Dibit und A zulässiges Vorbit. Daraus erhält man durch logische Verknüpfung NAB (= „Nur AB“) und NBC (= „Nur BC“): $NAB = ABG \& \text{ /BCG}$; $NBC = \text{ /ABG} \& BCG$. Wegen $NAB \& NBC = ABG \& \text{ /BCG} \& \text{ /ABG} \& BCG = 0$ sind die beiden Ausgangswerte NAB, NBC disjunkt (schließen sich gegenseitig aus) und kennzeichnen somit eindeutig, ob ausschließlich AB ($NAB = 1$) oder ausschließlich BC ($NBC = 1$) ein Dibit des entsprechenden Leitungscodes enthält.

Die Boolesche Funktionstabelle enthält die Werte für den CMI-Code. Beispielsweise ergibt sich bei $(A, B, C) = (0, 0, 0)$ für $ABG = 1$, denn $(A, B) = (0, 0)$ ist beim CMI-Code ein zulässiges Dibit und $C = 0$ ein zulässiges Folgebit, es könnte ein Ausschnitt der Folge $(0, 0) (0, 1)$ sein, die bei Codierung der binären Eingangsfolge 10 auftreten könnte. Gleichzeitig ist dann $BCG = 0$, denn ein Vorbit $A = 0$ und ein nachfolgendes Dibit $(B, C) = (0, 0)$ kann beim CMI-Code niemals auftreten. Also ist $NAB = 1 \& \text{ /0} = 1$ und $NBC = \text{ /1} \& 0 = 0$. Resultierend ergibt sich, daß $(A, B, C) = (0, 0, 0)$ nur ein Ausschnitt einer CMI-Folge sein kann, deren Dibit-Beginn bei A liegt. Dies kann für jede der acht möglichen Kombinationen von (A, B, C) ebenso ermittelt werden. Die Boolesche Funktionstabelle zeigt, daß jeweils drei Kombinationen eindeutig die Aussage „Dibit-Beginn bei A“ ($NAB = 1$) oder „Dibit-Beginn bei B“ ($NBC = 1$) ergeben.

Synchronisations-Schaltwerk

Die aus ABG, BCG erzeugten Binärsignale NAB, NBC werden für die Steuerung des Synchronisations-Schaltwerks verwendet. Mit diesem Schaltwerk (andere Bezeichnungen: Finite State Machine, FSM, oder Endlicher Automat) läßt sich die sich für eine empfangene Binärfolge ergebende Sequenz der binären Werte NAB bzw. NBC auswerten und damit eine eindeutige Syn-

chronisierung durchführen. Ist beispielsweise „Dibit-Beginn bei A“, dann ist häufig $NAB = 1$ und niemals $NBC = 1$. Durch Bitfehler infolge von Störungen gilt dies nicht mehr exakt. Allerdings ist selbst bei einer für reale Anwendungen extrem hohen Schrittfehlerrate von $p = 10^{-2}$ im Digitalkanal der mittlere Abstand zweier Bitfehler $N = 1/p = 10^2 = 100$, also 50 Dibit. Das Synchronisations-Schaltwerk ermittelt aus den Sequenzen der beiden binären Eingangssignale NAB und NBC, welches Registerpaar des Parallelregisters (entweder AB oder BC) tatsächlich das zu decodierende Dibit enthält. Das Synchronisations-Schaltwerk erzeugt für diese Auswahl das Ausgangssignal SEL (von SElect). Falls eine Anzeige des Synchronisier-Zustands erwünscht ist, läßt sich zusätzlich ein Ausgangssignal SYN (von SYNchron) erzeugen.

Die Funktion eines Schaltwerks läßt sich eindeutig durch ein Automatendiagramm definieren. *Bild 6* zeigt das Zustands-Übergangs-Diagramm eines „Moore-Automaten“ für diese Aufgabe, nachfolgend kurz „Automatendiagramm“ genannt; vgl. hierzu auch [4, 5, 6]. Bei der Interpretation des Automatendiagramms ist folgende (übliche) Vereinbarung zu beachten: Ist keine der von einem Zustand wegführenden Übergangsbedingungen erfüllt, verbleibt der Automat in diesem Zustand; der zugehörige „Übergang“ ist nicht dargestellt. Dies erspart im Automatendiagramm das Einzeichnen einer „Schlinge“ (Kante von einem Knoten zum selben Knoten) mit der Übergangsbedingung „REST“ (nämlich alle „REST-Kombinationen“ der Eingangsvariablen, die nicht in den vom betrachteten Zustand wegführenden Übergangsbedingungen auftreten).

Das Automatendiagramm für diese Aufgabe hat eine einfache, lineare Struktur, ähnlich einem Vorwärts-/Rückwärts-Zähler. Die Ausgabewerte sind in der Reihenfolge SEL, SYN in den Zustandsknoten eingetragen.

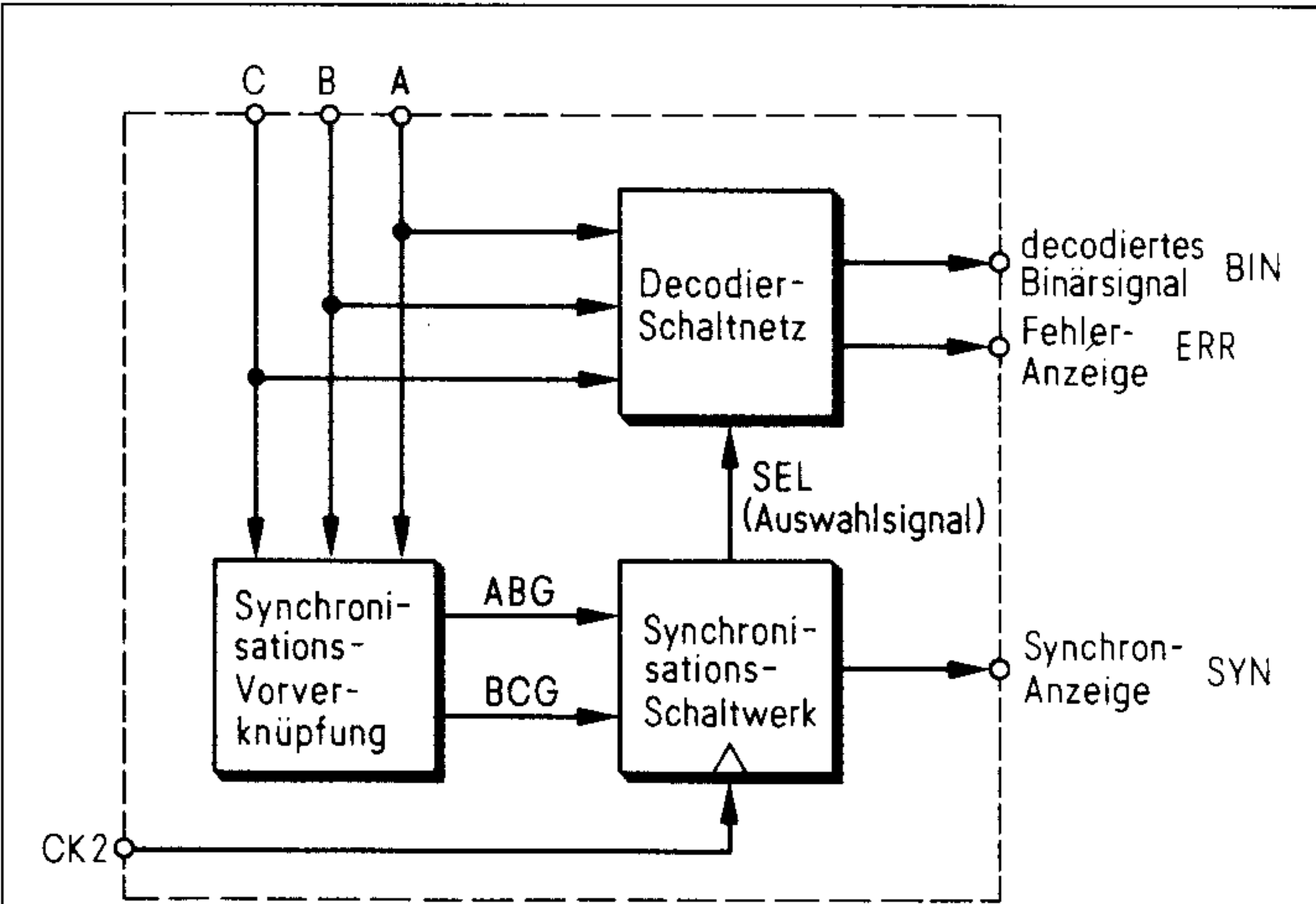


Bild 5. Prinzip des Decodier-Logikblocks. Bei Fehlern nimmt ein dafür vorgesehener Ausgang H-Potential ein. Die Synchron-Anzeige ist nicht dringend notwendig.

Ein Eingangssignal $NAB = 1$ ergibt stets einen Übergang zum linken Nachbarknoten, ein Eingangssignal $NBC = 1$ stets einen Übergang zum rechten Nachbarknoten. Ist ein Wechseln nach links oder rechts nicht möglich (im linken oder rechten Endzustand) oder liegt $(NAB, NBC) = (0, 0)$ am Eingang an, bleibt der gegenwärtige Zustand erhalten (vgl. obige Bemerkung).

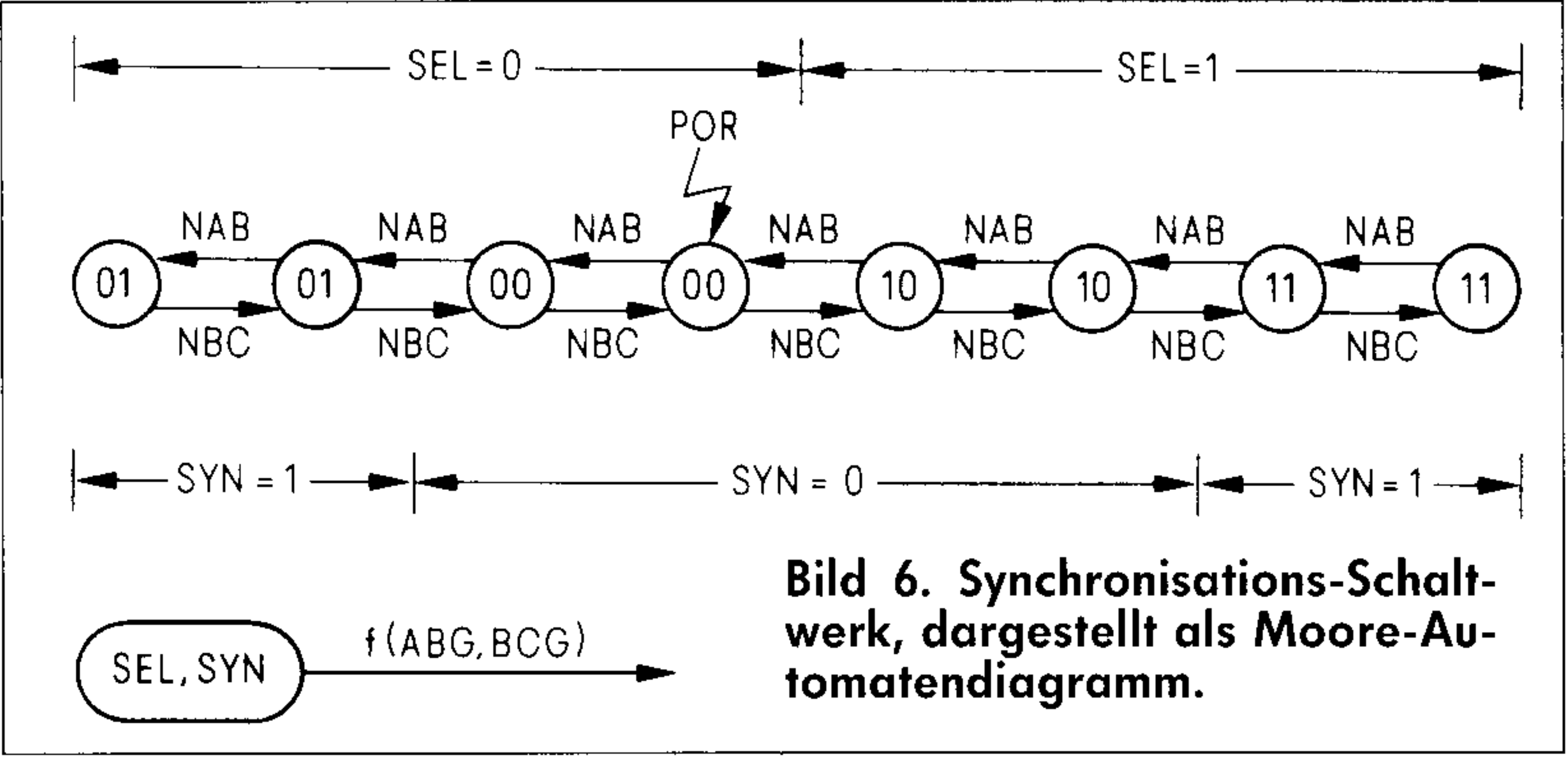
Startzustand sei der links von der Mittellinie gelegene Zustand, gekennzeichnet durch das Blitzsymbol mit der Bezeichnung POR (Power On Reset). Es erfolgt jetzt ein „Aufsynchronisieren“ auf die Dibit-Sequenz. Dazu wird geprüft, welche Werte die beiden disjunkten (aus ABG, BCG gebildeten) Verknüpfungssignale NAB, NBC aufweisen. Mit mehreren Werten $NAB = 1$ („Nur AB enthält gültiges Dibit“) gelangt man in die linke Endposition des Automatendiagramms, mit mehreren aufeinanderfolgenden Werten von $NBC = 1$ („Nur BC enthält gültiges Dibit“) in die rechte Endposition.

Mit dem Ausgangssignal SEL wird die Decodiertabelle im Block „Decodier-Schaltnetz“ ausgewählt. $SEL = 0$ erzwingt die Decodierung von AB, $SEL = 1$ erzwingt die von BC. Deswegen ist SEL für die gesamte linke Hälfte des Automatendiagramms 0, für die gesamte rechte Hälfte 1.

SYN zeigt lediglich nach außen den Synchronzustand an, für die Schaltungsfunktion ist die Erzeugung der Ausgangsgröße SYN nicht erforderlich. Sie kann zur Ansteuerung einer LED für die Anzeige des Synchronzustandes dienen. Bei elektrischer Übertragungstechnik auf symmetrischen Leitungen ist beispielsweise ein „Verpolen“ der beiden Adern daran zu erkennen, daß kein „Aufsynchronisieren auf die Dibit-Folge“ möglich ist. Bei optischer Übertragungstechnik oder Verwendung von Koaxialleitungen kann kein Verpolen eintreten. Falls keine Synchronisationsanzeige erforderlich ist, kann das Ausgangssignal SYN entfallen. Hier ist SYN für die beiden linken und die beiden rechten Randzustände des Automatendiagramms auf 1 gesetzt. Bei Erreichen dieser Zustände ist ein Aufsynchronisieren auf die Dibit-Sequenz erfolgt. Für alle restlichen Zustände ist SYN auf 0 gesetzt. Diese Zustände werden nur während des Synchronisiervorgangs nach dem Einschalten durchlaufen, bei Übertragungssystemen mit umschaltbarer Leitungscodierung (wie dem hier zugrundeliegenden Laborsystem) auch nach dem Umschalten des Leitungscodes.

Nach dem Aufsynchronisieren nimmt das System den linken oder rechten Randzustand an. Ein isoliert auftretender Bitfehler kann dann einen Wechsel zum Nachbarzustand bewirken, den es danach gleich wieder (da der Bitfehler isoliert auftrat) in Richtung Randzustand verläßt. Isolierte Bitfehler bleiben damit ohne jede Auswirkung auf die Synchronisierung.

Dieses Automatendiagramm stellt eine mögliche Lösung dar, andere ähnliche sind ebenso denkbar. Sind



beispielsweise nur extrem kleine Bitfehlerraten und keine Büschelstörungen zu erwarten, kann man mit insgesamt vier Zuständen auskommen.

Decodier-Schaltnetz

Bei bestehendem Dibit-Synchronismus kann die weitere Decodierung durch eine reine Verknüpfungsschaltung (Decodier-Schaltnetz) erfolgen. *Tabelle 3* zeigt die Boolesche Funktionstabelle für den Fall der CMI-Decodierung, sie ergibt sich entsprechend den Codierregeln des verwendeten Leitungscodes. Beispielsweise zeigt Zeile 1, daß bei $SEL = 0$ (also AB enthält das zu decodierende Dibit) unabhängig vom Wert von C die Kombination $(A, B) = (0, 0)$ zu 1 decodiert wird. Unzulässige Kombinationen des Leitungscodes können wahlweise zu 0 oder 1 codiert werden, in der Funktionstabelle ist dies durch einen Querstrich dargestellt. Damit kann der Logik-Compiler die logischen Terme optimieren. Das Ausgangssignal ERR (von ERRor) ist 1, wenn ein nach Codiervorschrift unzulässiges Dibit auftritt, ansonsten 0. Für die Schaltungsfunktion ist die Erzeugung von ERR nicht nötig, es kann zur Fehleranzeige dienen.

Für die obige Beschreibung ist stets der CMI-Code zugrunde gelegt. Beim MAN-Code ändern sich lediglich die Booleschen Funktionstabellen für die Synchronisations-Vorverknüpfung und das Decodier-Schaltnetz: das Automatendiagramm für das Synchronisations-Schaltwerk läßt sich unverändert verwenden. Die für den MAN/BIN-Leitungsdecodierer gültigen Booleschen Funktionstabellen zeigen die *Tabellen 4* und *5*. Die NAB-, NBC-Funktionstabellen für den MAN-Leitungs-

Tabelle 3. Boolesche Funktionstabelle für das CMI/BIN-Decodier-Schaltnetz

| A | B | C | SEL | BIN | ERR |
|---|---|---|-----|-----|-----|
| 0 | 0 | – | 0 | 1 | 0 |
| 0 | 1 | – | 0 | 0 | 0 |
| 1 | 0 | – | 0 | – | 1 |
| 1 | 1 | – | 0 | 1 | 0 |
| – | 0 | 0 | 1 | 1 | 0 |
| – | 0 | 1 | 1 | 0 | 0 |
| – | 1 | 0 | 1 | – | 1 |
| – | 1 | 1 | 1 | 1 | 0 |

A, B, C = Bezeichnung aufeinanderfolgender Dibit-Elemente
 SEL = Auswahlsignal des Synchronisations-Schaltwerks
 BIN = Decodiertes Bit
 ERR = Fehler-Anzeige (unzulässiges Dibit)
 – = beliebiger Binärwert

code weisen weniger „Einsen“ auf als die entsprechenden Tabellen für den CMI-Leitungscode. Das Aufsynchronisieren auf die Dibit-Folge erfolgt deshalb hier etwas langsamer.

Schaltungstechnische Realisierung

Durch die vorstehende vollständige Beschreibung der Gesamtlogik mit Booleschen Funktionstabellen und Automatendiagrammen ist der logische Schaltungsentwurf abgeschlossen, er ist nun in der Beschreibungssprache eines Logiksynthese-Softwarepakets zu formulieren. Ausgehend von diesem Quelltext erzeugt ein derartiges Programm eine standardisierte Datei, beispielsweise im sog. „JEDEC“-Format, mit dem mittels eines Programmiergerätes ein geeigneter PLD-Logikbaustein programmiert werden kann.

Bei einem Asic kann dieser logische Entwurf unmittelbar physikalisch realisiert und mit anderen Teilschaltungen integriert werden. Bei einfachen (und damit billigen) PLD-Bausteinen ist zu bedenken, daß diese nur einen Takteingang haben und sich daher alle Register nur mit demselben Taktsignal takten lassen. Beim beschriebenen Schaltungskonzept muß das serielle Binärsignal in das Schieberegister mit dem Taktsignal CK1 eingelesen werden, die Übernahme in das Auffangregister und die weitere Verarbeitung im Logikblock müssen jedoch mit dem Taktsignal CK2 mit halber Taktfrequenz erfolgen. Dies bedeutet für die schaltungstechnische Realisierung, daß beispielsweise:

- ein Standard-IC für die Serien-Parallel-Umsetzung (z.B. das 8-Bit-Schieberegister mit Auffangregister MM74HC595) und ein einfacher PLD-Baustein Einsatz finden, oder aber
- ein PLD mit zwei unabhängig taktbaren Registergruppen eingesetzt wird, so daß sich die beiden mit verschiedener Taktfrequenz arbeitenden Schaltungsblöcke in einem PLD integrieren lassen.

Bei der physikalischen Realisierung für das hier zugrundeliegende Laborsystem zur Digitalsignal-Übertragung wurde aus Kostengründen die erste Lösung verwendet.

Der oben beschriebene „logische Entwurf“ für einen synchronen CMI-/MAN-Leitungsdecoder war Vorgabe für eine physikalische Realisierung im Rahmen zweier Diplomarbeiten [8, 9]. Dort erfolgte die Formulierung der Booleschen Funktionstabellen und des Automatendiagramms als Quelltext für das Logiksynthese-Softwarepaket LOG/iC [7]. Für die physikalische Realisierung wurden aus Kostengründen ein getrennter 8-Bit-Schieberegisterbaustein (siehe oben) und ein einfaches PLD verwendet. Die Leitungsdecodierung ist dabei umschaltbar ausgelegt, so daß sich sowohl CMI- als auch MAN-Leitungssignale decodieren lassen.

Das Logiksynthese-Softwarepaket LOG/iC erlaubt einen „blockstrukturierten Entwurf“, die oben definierten Funktionsblöcke der Schaltung treten dabei direkt als „Blöcke“ des LOG/iC-Quelltextes auf. Der Quelltext ist (bis auf die Angabe der Anschlußbelegung) unabhängig vom verwendeten programmierbaren Logikbaustein. Dies ermöglicht den Einsatz beliebiger PLD-Bau-

Tabelle 4. Boolesche Funktionstabelle der Synchronisations-Vorverknüpfung für den MAN-Leitungscode

| A | B | C | ABG | BCG | NAB | NBC |
|---|---|---|-----|-----|-----|-----|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |

A, B, C = Bezeichnung aufeinanderfolgender Dibit-Elemente
ABG = ABGültig NAB = ABG & /BCG
BCG = BCGültig NBC = BCG & /ABG

Tabelle 5. Boolesche Funktionstabelle für das MAN/BIN-Decodier-Schaltnetz

| A | B | C | SEL | BIN | ERR |
|---|---|---|-----|-----|-----|
| 0 | 0 | – | 0 | – | 1 |
| 0 | 1 | – | 0 | 0 | 0 |
| 1 | 0 | – | 0 | 1 | 0 |
| 1 | 1 | – | 0 | – | 1 |
| – | 0 | 0 | 1 | – | 1 |
| – | 0 | 1 | 1 | 0 | 0 |
| – | 1 | 0 | 1 | 1 | 0 |
| – | 1 | 1 | 1 | – | 1 |

A, B, C = Bezeichnung aufeinanderfolgender Dibit-Elemente
SEL = Auswahlsignal vom Synchronisations-Schaltwerk
BIN = Decodiertes Bit – = beliebiger Binärwert
ERR = Unzulässiges Dibit

steine (ausreichender Schaltungskomplexität und Taktfrequenz) beliebiger Hersteller.

Ausgehend vom oben vollständig beschriebenen logischen Entwurf kann die physikalische Realisierung natürlich auch mit einem anderen geeigneten Logiksynthese-Softwarepaket und mit jedem programmierbaren Logikbaustein ausreichender Schaltungskomplexität und Taktfrequenz erfolgen.

Der Autor wurde bereits in Heft 12/91 vorgestellt.

Literatur

[1] Hartl, P.: Fernwirktechnik der Raumfahrt. Springer-Verlag, Berlin 1988.
[2] Söder, G.; Tröndle, K.: Digitale Übertragungssysteme. Springer-Verlag, Berlin 1985.
[3] Morgenstern, G.; Wellhausen, H.-W.: Drei- und mehrstufige Leitungscode für die Digitalsignal-Übertragung. Der Fernmelde-Ingenieur, 41. Jahrgang, 1987, H. 3.
[4] Nocker, R.; Hedke, L.: Leitungscodierung mit PAL-Baustein; CMI-, MAN- und AMI-Code mit einer Schaltung erzeugbar. ELEKTRONIK 1990, H. 23, S. 66...73.
[5] Nocker, R.; Hedke, L.: HDB3-Leitungscodierer mit programmierbarem Logikbaustein. Synthese anhand von Automatendiagrammen. ELEKTRONIK 1991, H. 12, S. 74...79.
[6] Tietze, U.; Schenk, Ch.: Halbleiter-Schaltungstechnik. Springer-Verlag, Berlin 1978.
[7] ISDATA GmbH. Programmsystem LOG/iC (Release 3.2) Syntax- und Compiler-Handbuch. Karlsruhe 1989.
[8] Domfeld, T.; Tetzlaff, U.: Regenerativverstärker für CMI/MAN-codierte Binärsignale. Diplomarbeit 1990, FH Hannover, FG Kommunikationstechnik (Betreuer: Prof. Dr.-Ing. R. Nocker).
[9] Meyer, F.; Polkahn, M.: Infrarot-Digitalsignal-Übertragungssystem 128 kBd. Diplomarbeit 1991, FH Hannover, FG Kommunikationstechnik (Betreuer: Prof. Dr.-Ing. R. Nocker).